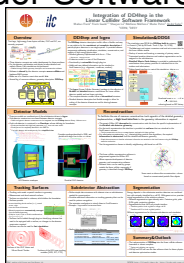


Integration of DD4hep in the Linear Collider Software Framework

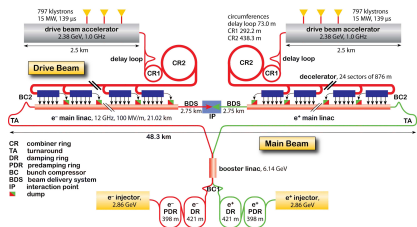


Markus Frank* Frank Gaede*,[†] Shaojun Lu[†]
 Nikiforos Nikiforou* Marko Petric* André Sailer*

*CERN, [†]DESY

CHEP 2015, Okinawa Institute of Technology, Okinawa, Japan
 April 17, 2015

Two large, high-energy linear lepton colliders, CLIC and ILC, are currently proposed (See Yamauchi-san's opening words, or come to the Linear Collider Workshop next week at KEK)

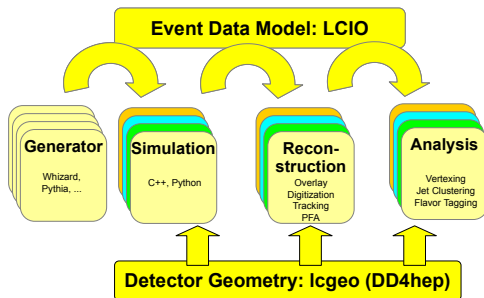


- Three detector concepts (ILC: SiD, ILD; CLIC) are under development for these machines
- For the detector optimisation and physics studies very detailed and realistic Monte Carlo simulations are performed

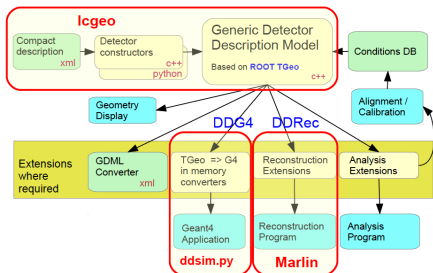
Overview



- Detector optimisation studies are enabled through flexible geometry description and reconstruction software
- Software is **shared** by the detector concepts **across colliders** and hardware R&D groups
- Make use of a common *event data model*: **lcio** (F. Gaede et al., CHEP'03)
- Second leg of common software, *geometry description*: DD4hep (M. Frank et al. DD4hep, CHEP'13)

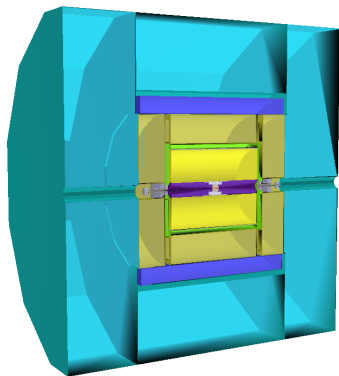


DD4hep in a Nutshell

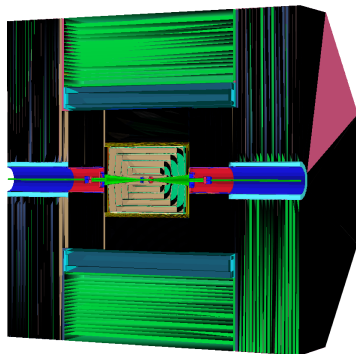


- DD4hep: Detector Description for HEP, one source of geometry, easy-to-use and generic, detector geometry for the full experiment life-cycle
- The **LCGEO** (Linear Collider Geometry) package is the collection of flexible but detailed detector constructors for **all** linear collider **detector models** and test beam activities

Example Detector Geometries



ILD Detector envelopes

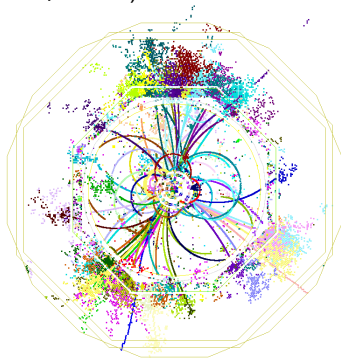


Detailed CLIC detector

Simulation with DDG4



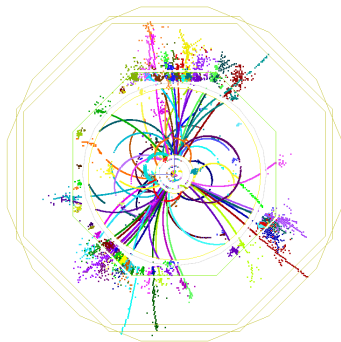
- The simulation uses DDG4 the DD4hep **built-in** gateway to GEANT4 (Talk by M.Frank: Track 2, Apr. 16, 11:45)



Simulated event $e^+e^- \rightarrow t\bar{t}$ at
 $\sqrt{s} = 500$ GeV:
colouring based on MC history

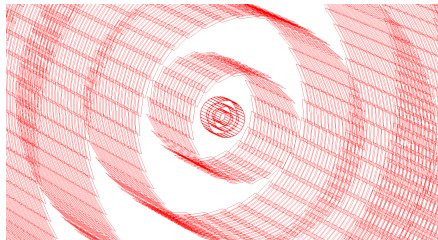
To facilitate the use of common reconstruction tools agnostic of the detailed geometry implementation, a **high level interface** to the geometry information is required

- The generic DDRec API **decouples** the **reconstruction** code from the specific implementation of the **detailed sub-detector geometry**
- Use surfaces for track reconstruction
- Simple data structs to contain abstract view of geometry required for reconstruction

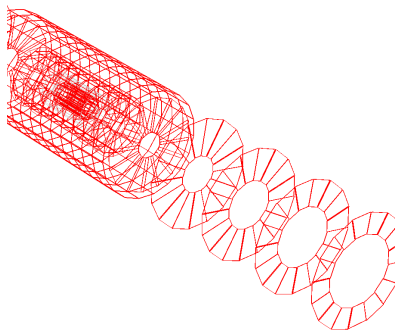


Event after reconstruction: colours based on reconstructed particle flow objects

- Surfaces can be added through plug-ins, **no changes** to the geometry implementation **required**
- Surfaces can also be used for **fast simulation**



Surfaces of the CLIC Vertex and
Silicon Barrel Tracker



Surfaces of the ILD inner tracking
modules (VXD, SIT, FTD)

どうもありがとうございます



Many thanks

- For your attention
- To the organisers for a great conference

Maybe we can all be back to Japan soon for the groundbreaking of the ILC